BCS Digital Industries Apprenticeship

Standard Specific Guidance for Training Providers

Level 4 Software Developer Apprenticeship

Version 3.0
May 2018

Change History

Any changes made to the project shall be clearly documented with a change history log. This shall include the latest version number, date of the amendment and changes made. The purpose is to identify quickly what changes have been made.

Version Number and Date	Changes Made
November 2016 V1.0	Document created from three earlier documents: • Training Provider Reference & Guide • Summative Portfolio Guide • Employer Reference Guide
November 2016 V1.1	Summative portfolio declaration included within Template 5
February 2017 V1.2	SFIA mappings updated to be more focused on the standard and occupational brief competencies and detailed descriptions.
March 2017 V1.3	Proofread final version
January 2018 V2.0	Update to technical competencies, knowledge standards and work activities.
May 2018 V3.0	Removal of typical evidence and update to work activities.

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Contents

Purpose of this Document

The purpose of this document is to provide useful information and suggested supporting documentation specific to the Software Developer Apprenticeship. It should be read in conjunction with the BCS General Guidance for Apprentices, Employers and Training Providers document and is designed to give training providers some tools to help them build their own programme from training plan through to end point assessment.

This guide will provide:
- supporting information around how help to the software developer apprentice meet and go beyond the standard;
- a number of useful documents to support the training provider in meeting their responsibilities in managing the apprenticeship from training plan through to the end point assessment;
- evidence checklists to help the training provider support the apprentice in completing their summative portfolio;
- a template for completing the employer reference.

Introduction

The BCS Level 4 Software Developer Apprenticeship is one of the suite of Digital Industries Apprenticeships that have been designed by the industry to address skills shortages and meet the ever-changing needs of UK employers.

The General Guidance for Apprentices, Employers and Training Providers provides the broad view on how to run an apprenticeship programme to the BCS Digital Industries Standard. The collection of tables and templates contained within this document has been designed to give training providers the tools to build their programme and to assist them in helping apprentices and employers towards the successful completion of each element of the end point assessment.

The areas where a training provider should be involved in ensuring a successful outcome to the apprenticeship are:

- mapping and assessing work against the standard;
- advising the employer and the apprentice on which knowledge modules, vendor or professional certificates and other relevant training and activities are most appropriate for their requirements, and agree a suitable training plan;
- assisting the apprentice with applying knowledge in the workplace;
- acting as an advisor to the apprentice and the employer to ensure the programme remains on track and any concerns are addressed;
- helping the apprentice to select evidence for their summative portfolio;
- supporting the apprentice through the synoptic project;
- confirming the apprentice's readiness for the end point assessment.

The following series of checklists can be used by the training provider to help manage the process through to completion. Training providers may substitute their own processes and documentation as they see fit in order to effectively manage their key areas of responsibility as set out above.

The Software Developer Apprentice

The primary roles of a software developer are to:

- build and test simple, high-quality code across front end, logic and database layers;
- working as part of a larger team, in which they will have responsibility for some of the straightforward elements of the overall project;
- interpret design documentation and specifications.

Job titles may be different across different organisations so the role may also be referred to as web developer, application developer, mobile app developer, games developer or software developer.

Business Proficiencies

The proficiencies that should be demonstrated by an apprentice in software development are listed below.

Business skills
- The apprentice can demonstrate an analytical and systematic approach to issue resolution.
- The apprentice takes the initiative in identifying and negotiating appropriate personal development opportunities.
- The apprentice can demonstrate effective communication skills.
- The apprentice can contribute fully to the work of teams.
- The apprentice plans, schedules and monitors own work (and that of others where applicable) competently within limited deadlines and according to relevant legislation, standards and procedures.
- The apprentice can appreciate the wider business context, and how their role relates to other roles and to the business of the employer or client.

Complexity
- The apprentice can perform a range of work, sometimes complex and non-routine, in a variety of environments.
- The apprentice can apply a methodical approach to issue definition and resolution.
- The apprentice undertakes all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools.

Autonomy
- The apprentice can operate under general direction.
- The apprentice uses discretion in identifying and responding to complex issues and assignments.
- The apprentice can receive specific instructions and has work reviewed at frequent milestones.
- The apprentice can determine when issues should be escalated to a higher level.

Influence
- The apprentice interacts with and influences colleagues.
- The apprentice has working level contact with customers, suppliers and partners.

- The apprentice may supervise others or make decisions which impact the work assigned to individuals or phases of projects.
- The apprentice makes decisions which influence the success of projects and team objectives.

Knowledge Standards, Technical Competence and Behaviour and Relationship Standards

Tables 1, 2 and 3 contain details of the topics that the training provider may decide to cover in their development plans and scheduled work activities in order to stretch the apprentice.

Table 1 – Software Developer – Knowledge Standards

The knowledge standards define learning that must take place during the apprenticeship, **both through the activities and the apprentice's own independent learning**. The additional learning outcomes detailed in the table show how a training provider can stretch the apprentice's learning beyond the requirement as set out in the occupational brief. However, it is important to remember that stretching the apprentice in this way will only have a bearing on their final grading if the impact is demonstrated through their competence in the end point assessment. These knowledge standards, therefore, show the additional learning that may support the apprentice in improving their overall competence. Technical knowledge and understanding is assessed throughout the apprenticeship through a combination of Ofqual regulated knowledge modules and specified vendor and professional qualifications. These must be passed before the end point assessment can take place.

Knowledge standard	Expected requirement	Suggested learning outcomes to meet the standard and stretch the apprentice to exceed the minimum requirement
Understands and operates at all stages of the software development life cycle.	What is the software development lifecycle (SDLC). What are the seven generic stages and their high-level deliverables from each stage. What are the main activities in each stage.	Explain the role and function of the system development lifecycle. Relate the seven generic stages of the software development lifecycle: - feasibility study; - requirements analysis; - design; - development; - testing; - implementation; - maintenance. illustrate the main activities in each stage of the software development lifecycle. Demonstrate the high-level deliverables from each stage of the software development lifecycle.
Understands the similarities and differences (taking into account positives and negatives of both approaches) between agile and waterfall software development methodologies.	What is the agile development method. What is the waterfall development method.	Describe the primary differences between the waterfall and the agile software development methods. Explain the respective strengths and weaknesses of each of the waterfall and agile software development methods.

	What are the strengths and weaknesses of both approaches.	Demonstrate the selection criteria for using either the waterfall or the agile software development method.
Understands how teams work effectively to produce software and contributes appropriately.	What are the roles that need to be undertaken. How are these roles distributed across a team. What team-working aspects are needed to ensure effective delivery of projects.	Relate the roles and responsibilities within software development and implementation (for example, analysts, designers, developers, testers and technical architects). Illustrate the structure of a software development team within an organisation. Describe the responsibilities of project managers in software development and the difference between project life cycles and software development life cycles. Demonstrate what team-working aspects are needed to ensure effective delivery of software projects.
Understands and applies software design approaches and patterns and can interpret and implement a given design, compliant with security and maintainability requirements.	Software design approaches. Software patterns. Documenting software designs. Secure development. Designing for software maintainability and re-use.	Demonstrate the purpose of software design: - to aid communication between actors; - as a basis for rigorous development; - to provide a standard approach; - to ensure consistency across the development; - to assist in the identification of reuse; - to compare the current situation with the required situation. Demonstrate the use of software patterns in the software design process. Explain the rationale for separating functional and non-functional requirements and understand how software designs can be documented and how the design documents will be used to support software implementation. Illustrate the need for secure development and give examples of how this can be included within the software implementation process. Discover the need for software maintainability and how software can be implemented in a manner that enables reuse and maintainability.
Understands and applies the maths required to be a	Understanding of basic algorithmic processing to	Contrast the four key techniques of computational thinking: decomposition, pattern recognition, abstraction, algorithms.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

software developer (e.g. algorithms, logic and data structures).	define the problem and / or the solution. Elements of programming logic - variables; assignment statements; data types; conditionals; loops; arrays; and input / output, Knowledge of at least two data structures – such as Arrays or Collection Classes.	Demonstrate how algorithms are used to create a logical solution to a computable problem: • the use of semi-formal specification of algorithms, based on a simplified computer model; • sources of algorithms; • development of code from an algorithm. Apply the primary elements of programming languages and their function: • types: numeric and non-numeric, elementary and derived, subtypes, and expressions such as assignments and input / output; • control structures: selection and iteration subprograms, procedures and functions; • data structures: arrays (1 and 2 dimensional), implementation of queues, stacks and lists. • concept of data abstraction.

These expected requirements are met through the delivery of the BCS Ofqual regulated knowledge modules and vendor training courses, details of which are contained in the course syllabi.

Table 2 – Software Developer – Technical Competency Standards

The competency standards have been defined to demonstrate that the knowledge learnt has been applied in real work tasks, activities and projects in a business environment.

Competencies are assessed throughout the apprenticeship through a combination of the employer reference, the synoptic project and a summative portfolio completed by apprentices from records of the work activities in which they have been involved.

The training provider should assist the employer to identify suitable work tasks, activities and projects within the scope of their normal business activities for the apprentice to practice what they have learnt and to demonstrate the competencies below.

The BCS Apprenticeship is mapped to an internationally recognised skills framework and to work activities in which a software developer apprentice would be involved.

The following table sets out these competencies and the expected requirements against the work activities that might be demonstrated at and beyond the minimum expectation. The format is explained below.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
This column contains the competency as it is listed in the apprenticeship standard.	*This column shows the expected requirements listed in the occupational brief for a successful outcome.*	*This column shows recognised work activities that demonstrate that the apprentice is meeting the expected requirement.* *The apprentice should be able to demonstrate all of these activities.*

The software developer competency standard, requirements and activities demonstrating competence follow.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Logic Writes good quality code (logic) with sound syntax in at least one language.	Apprentices can write code to achieve the desired functionality and which is easy to read and understand, with good naming, indentation and commenting, and applying the fundamentals of good coding: • development paradigms (where this is object-oriented programming this must include inheritance, abstractions, encapsulation, polymorphism) • software programming languages • software development tools (IDEs) • writing programs and methods • language-specific idioms • logic and flow-of-control Apprentices can apply: • Elements of programming – variables, assignment statements, data types, conditionals, loops, arrays and input / output. • Functions – modular programming dividing a program into components that can be independently debugged, maintained and reused; writing at least two reusable functions. • Algorithms and data structures – classical algorithms for sorting and searching, and fundamental data structures.	Designs moderately complex programs and program modifications from supplied specifications, using agreed standards and tools, to achieve a well-engineered result. Creates, amends and keeps track of moderately complex programs in accordance with the design. Documents all work in accordance with agreed standards. Takes part in reviews of own work. Takes part in reviews of the work of colleagues.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
User interface Can develop effective user interfaces for at least one channel.	Apprentices can apply the fundamental concepts of human–computer interaction or user experience design, the development practices leading to a high-quality user interface, and the programming techniques required to construct a graphical user interface (UI). Apprentices can interact with screen or UI designers to ensure the logic layer integrates with the user interface. Apprentices can develop user interface coding and implementation – techniques for building user interfaces – for at least one channel. Apprentices can interact with testers to optimise the user interface.	Designs simple applications using templates and tools to specify user/system interfaces, including for example: menus, screen dialogues, wireframes, boned rigs, inputs, reports, validation and error correction procedures, and processing rules. Contributes to detailed designs including for example: user interface (including colour/language/presentation/input methods/error handling and responses), user documentation, program specifications, and backup, recovery and restart procedures. Documents all work using required standards, methods and tools, including prototyping tools where appropriate. Constructs, interprets and executes test plans to verify accessibility and usability of completed systems.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Data Can effectively link code to the database / data sets.	Apprentices can link to a range of database types and embed data queries within their code. Apprentices can: • make a connection to a database • execute CRUD statements on the database • use one-off queries and stored procedures • transform returned data in to format the application requires	Within a project environment, assists in the investigation of application data and process requirements, documenting them according to the required standards, utilising the prescribed methods and tools. Within a project environment, applies data analysis and data modelling techniques, based upon a general understanding of the business process, to establish, modify or maintain a data structure and its associated components (e.g. entity descriptions, relationship descriptions, attribute definitions). Creates, amends and keeps track of moderately complex programs in accordance with the design.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Test Can test code and analyse results to correct errors found using either V-model manual testing and/or using unit testing.	Apprentices can test and analyse their code to identify errors as soon as possible in the coding process and on an interactive basis**.** Apprentices can: • apply test and debugging strategies • design and develop manual or unit tests • test code segment functionality against requirements • assess test results against expected results and acceptance criteria	Plans, designs and conducts tests of moderately complex programs; corrects errors and retests to achieve an error-free result. Reviews requirements and specifications, and defines test conditions. Analyses test requirements, designs and builds simple test case suites, test scripts and test procedures, with expected results. Interprets and executes sets of moderately complex test scripts using agreed methods and standards, recording and reporting outcomes. Checks test results, and documents test failures and successes compared with predetermined criteria, in accordance with agreed standards. Analyses and reports test results to supervisor and/or other colleagues in a clear and concise manner. Identifies and reports issues and risks associated with own work. Reviews and tests non-functional aspects of systems at a high-level.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Problem solving Can apply structured techniques to problem solving, can debug code and can understand the structure of programs in order to identify and resolve issues.	Apprentices can: • use a minimum of two problem-solving tools and techniques to identify and resolve programming issues • apply structured problem solving methods • apply problem-solving techniques to programming activities	Plans, designs and conducts tests of moderately complex programs; corrects errors and retests to achieve an error-free result. Carries out fault diagnosis relating to simple software failures, reporting the results of the diagnosis in a clear and concise manner.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Design Can create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards.	Apprentices can: • take a high-level design and can interpret and convert the design in to simple data models and/or program modules to communicate it to others • apply software design methodologies (e.g. structured or object-oriented) • use standard design notation such as UML • apply data modelling • apply reconcile design against analysis models • design software solutions to meet requirements	Designs moderately complex programs and program modifications from supplied specifications, using agreed standards and tools, to achieve a well-engineered result. Designs simple applications using templates and tools to specify user/system interfaces, including for example: menus, screen dialogues, wireframes, boned rigs, inputs, reports, validation and error correction procedures, and processing rules. Assists as part of a team on design of software components of larger systems. Produces components of detailed designs, such as: physical data flows, class diagrams, file layouts, common routines and utilities, program specifications or prototypes, and backup, recovery and restart procedures. Documents all work using required standards, methods and tools, including prototyping tools where appropriate.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Analysis Can understand and create basic analysis artefacts, such as use cases and/or user stories.	Apprentices can: • take a variety of data and business requirements and convert them in to basic analysis artefacts to understand and can clarify the intended use of the proposed software • identify and represent required functionality (e.g. use cases) • identify and represent activity workflow (e.g. activity diagrams)	Assists colleagues and clients/users to investigate and model business functions, processes, information flows and data structures, using various methodical and consistent techniques. Assists colleagues and clients/users in specifying information flows, processes/procedures and data objects that align with the needs of the business. Records work with appropriate documentation, meeting the required standards and uses suitable methods and tools. Understands the purpose and benefits of modelling, and uses established techniques as directed to model simple subject areas with clearly defined boundaries. Elicits and records business/context rules and concepts and confirms them with business experts. Develops models with input from subject matter experts and communicates the results back to them for review and confirmation. Reviews models with more experienced modellers for consistency, completeness and accuracy. May assist in more complex modelling activities.

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Deployment Can understand and utilise skills to build, manage and deploy code into enterprise environments.	Apprentices can package and build completed programs as appropriate to the resources available for deployment and for migration to different environments, including: • developing appropriate user documentation • planning for user training • data migration	Produces software builds, for loading onto target hardware, from software source code (typically held within a configuration management system). Conducts a series of tests as defined in an integration test specification and records the details of any failures in a concise but complete manner. Assists in the configuration of software and equipment for the systems testing of platform-specific versions of one or more software products. Participates in the configuration of software and equipment for systems testing of platform specific versions of one or more software products with minimum supervision.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Development life cycle Can operate at all stages of the software development lifecycle, with increasing breadth and depth over time with initial focus on build and test.	Apprentices can operate as software developers, showing a good understanding of the other phases of the software development lifecycle and the deliverables that are produced at each stage and as relevant to the development methodology (waterfall, agile, test led etc.) In addition to the stages above, the apprentice can also: - operate in the support and maintenance phases - advise third line support for relevant applications - fix bugs and deal with change requests	Provides input to known problem and solution and change management function. May carry out early life support activities such as providing support advice to initial users. Receives and logs requests for application support from help desk, other service delivery staff and/or users. Within own area of competence investigates issues and other requests for application support and determines appropriate actions to take. Within own area of competence and working closely with more senior colleagues, provides correct responses to requests for software application support by means of for example: making modifications to system parameters, developing work-arounds or site-specific enhancements, reconfiguring systems, changing operating procedures, training users or operations staff, producing additional documentation, or escalating requests to other software development staff or software suppliers. Ensures all work is carried out and documented in accordance with required standards, methods and procedures.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Applying Good Practices Can apply good practice approaches according to the relevant paradigm (for example object-oriented, event driven or procedural)	Apprentices can identify and follow standards and good practice that can improve programming efficiency, style and quality, including: • programming standards, both organisational and external • generic best practices, including readability, reusability, maintainability • best practice approaches of different paradigms and languages	Designs moderately complex programs and program modifications from supplied specifications, using agreed standards and tools, to achieve a well-engineered result. Documents all work in accordance with agreed standards. Conducts reviews of supplied specifications, with others as appropriate. Takes part in reviews of own work. Takes part in reviews of the work of colleagues.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Interpret and Follow Can interpret and follow software designs, functional/technical specifications, company defined coding standards or industry good practice for coding, testing frameworks and methodologies, and company, team or client approaches to continuous integration, version and source control.	Apprentices can adapt to the employer's domain and context for software development and interpret and follow the software development approach being implemented. Apprentices can: • read software designs and functional/technical specifications, especially those based upon the employer domain and context • identify, interpret and follow coding standards • identify, interpret and follow best practice coding approaches for specific paradigms and languages • identify, interpret and follow company, team or client approaches to continuous integration, version and source control	Designs moderately complex programs and program modifications from supplied specifications, using agreed standards and tools, to achieve a well-engineered result. Documents all work in accordance with agreed standards. Conducts reviews of supplied specifications, with others as appropriate. Interprets and executes sets of moderately complex test scripts using agreed methods and standards, recording and reporting outcomes. Interprets and executes sets of moderately complex test scripts using agreed methods and standards.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Respond to Business Environment Can respond to the business environment and business issues related to software development.	Apprentices can apply the following considerations when working on projects: • business context • business drivers (efficiency gains, increased functionality and improved quality of outputs)	Assists colleagues and clients/users to investigate and model business functions, processes, information flows and data structures, using various methodical and consistent techniques. Assists colleagues and clients/users in specifying information flows, processes/procedures and data objects that align with the needs of the business. Receives and logs requests for application support from help desk, other service delivery staff and/or users. Within own area of competence investigates issues and other requests for application support and determines appropriate actions to take.

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Industry Environment Can operate effectively in their own business's, their customers' and the industry's environments.	Can demonstrate working within operational requirements such as health and safety, budgets, brands and normal business protocols.	Undertakes all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools.

Competency standard	Expected requirement	Work activities demonstrating the minimum expected level of competence
Application of Mathematics Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures).	Can apply algorithms, logic and data structures.	Applies the maths required during Software Development Projects.

Criteria for demonstrating **Significantly** higher competencies.
Understands and applies a wide range of tools and methods
Accurately and appropriately applies and effectively implements the right tools and methods in a variety of different situations
A sophisticated user - fully exploits the functionality/capability of the tools and methods
Extensive and deep understanding of different tools and methods and how and why they can be applied in different contexts
Deals confidently and capably with a high level of interrelated and interdependent factors in their work

Table 3 – Generic Behaviour and Relationship Standards

The behaviour and relationship standards have been defined to demonstrate that the apprentice applies the good behaviours and interpersonal skills that are needed in a business environment. Behaviours and business relationship skills are assessed throughout the apprenticeship through a combination of the employer reference, the synoptic project and a summative portfolio, which is completed by apprentices from records of the work activities in which they have been involved. The training provider could assist the apprentice by offering some additional soft skills training over and above their apprenticeship. The apprenticeship standard sets out the attributes required within the occupational brief, which can be accessed via the Apprenticeship section of www.bcs.org.

Behaviour and relationship standards	Expected requirement
Apprentices can demonstrate the full range of skills, knowledge and behaviours required to fulfil their job role. Apprentices can demonstrate how they contribute to the wider business objectives and show an understanding of the wider business environment.	The job: • Understanding the scope of the job role • Knowing what skills, knowledge and behaviours are needed to do the job well • Being aware of their own strengths in the job role, and any areas for improvement • Appreciating who else is important, for them to do their job and fulfil the role effectively (e.g. colleagues, managers, other stakeholders) • Being aware of potential risks in the job role (e.g. security, privacy, regulatory) • Using personal attributes effectively in the role, e.g. entrepreneurship • Understanding how the job fits into the organisation as a whole • Knowing what the next steps in their career might be The organisation: • Understanding the goals, vision and values of the organisation • Knowing how they contribute to these in their own work • Being aware of the commercial objectives of the tasks / projects they are working on • Understanding the importance of meeting or exceeding customers' requirements and expectations • Being in tune with the organisation's culture

Behaviour and relationship standards	Expected requirement
	• Knowing how the organisation works, including its informal culture, internal networks etc. The environment: • Being aware of the position of the organisation in the economy and its contribution to society • Understanding the key external factors that shape the way the organisation function, e.g. regulation • Knowing how the organisation can gain advantage in the industry, e.g. through innovation, technology, customer service etc.
Apprentices can demonstrate the ability to use both logical and creative thinking skills when undertaking work tasks, recognising and applying techniques from both. Apprentices can show that they recognise problems inherent in, or emerging during, work tasks, and can tackle them effectively.	Logical thinking: • Understanding initial premise(s) and preconditions • Analysing situations from known facts • Recognising the conclusion to be reached • Proceeding by rational steps • Evaluating information, judging its relevance and value • Supporting conclusions, using reasoned arguments and evidence Creative thinking: • Looking at situations from a fresh perspective • Exploring ideas and possibilities • Making connections between different aspects • Questioning assumptions • Generating solutions that may be imaginative or unconventional • Devising new approaches • Adapting ideas and approaches as conditions or circumstances change Problem-solving: • Analysing situations • Defining goals • Developing solutions

Behaviour and relationship standards	Expected requirement
	• Prioritising actions • Dealing with unexpected occurrences
Apprentices can manage relationships with work colleagues, including those in more senior roles, customers / clients and other stakeholders, internal or external, and as appropriate to their roles, so as to gain their confidence, keep them involved and maintain their support for the task / project in hand. Apprentices can establish and maintain productive working relationships, and can use a range of different techniques for doing so.	Managing relationships: • Understanding the value and importance of good relationships • Adopting a way of working that maintains and improves relationships • Involving other people in decisions and actions • Influencing others by listening to and incorporating their ideas and views • Acknowledging other people's accomplishments and strengths • Overcoming barriers that prevent productive action • Managing conflict constructively • Promoting teamwork by encouraging others to participate Customer / client relationships: • Establishing contact with customers / clients and keeping in touch • Keeping customers/clients informed ('no surprises') • Understanding their requirements, including constraints and limiting factors • Setting reasonable expectations • Involving them in decisions and actions ('co-production') • Interacting positively with them Communicating in different ways • Providing a complete answer in response to queries ('transparency', 'full disclosure') • Actively seeking feedback Stakeholders: • Understanding who they are and what their 'stake' is • Prioritising stakeholders in terms of their importance, power to affect the task and interest in it • Using stakeholders' views to shape projects early on • Gaining support from stakeholders, e.g. to win resources • Agreeing objectives • Managing expectations

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Behaviour and relationship standards	Expected requirement
Apprentices can communicate effectively with a range of people at work, one-to-one and in groups, in different situations and using a variety of methods. Apprentices can demonstrate various methods of communication, with an understanding of the strengths, weaknesses and limitations of these, the factors that may disrupt it, and the importance of checking other people's understanding.	Intention / purpose: - Understanding the purpose of communicating in a particular situation or circumstance (e.g. inform, instruct, suggest, discuss, negotiate etc.) - Checking that the person / people with whom one is communicating also understand the purpose - Being sensitive to the dynamics of the situation - Being aware of anything that might disrupt the effectiveness of the communication (e.g. status, past history) Method: - Knowing the range of possible communication methods (e.g. spoken, written, graphical, multimedia) - Choosing a good, appropriate method for the situation - Being aware of the limitations of the chosen method, and the possible risks of miscommunication (e.g. ambiguity) - Taking account of the affective dimensions of the method (e.g. body language, tone of voice, eye contact, facial expression etc.) Execution: - Expressing oneself clearly and succinctly, but not oversimplifying - Checking that the other person/people understand what is being expressed - Taking account of the potential barriers to understanding (e.g. filtering, selective perception, information overload) - Modifying the purpose and methods of communication during a situation in response to cues from the other person /people

These attributes are difficult to measure and are subjective in nature so cannot actually guarantee that any greater level of competence or proficiency is being demonstrated. The BCS Apprenticeship is mapped to the Skills Framework for the Information Age (SFIA), an internationally recognised skills framework, and to observable activities that a software development apprentice working to the level of responsibility appropriate for the role should demonstrate. Accordingly, the proficiencies that should be demonstrated by an apprentice in software development are shown below.

Proficiency standard	Work activities demonstrating expected level of proficiency	Work activities demonstrating competence beyond the minimum expected
Business skills	Demonstrates an analytical and systematic approach to issue resolution. Takes the initiative in identifying and negotiating appropriate personal development opportunities. Works independently and takes responsibility. Demonstrates effective communication skills. Contributes fully to the work of teams. Plans, schedules and monitors own work (and that of others where applicable) competently within limited deadlines and according to relevant legislation, standards and procedures. Appreciates the wider business context, and how their role relates to other roles and to the business of the employer or client.	Selects appropriately from applicable standards, methods, tools and applications. Undertakes work that is more complex, more critical or more difficult. Demonstrates an ability to extend or enhance their approach to work and the quality of outcomes. Doesn't just solve the problem but explores all known options to do it better, more efficiently, more elegantly or better meet customer needs. Shows good project management skills, in defining problem, identifying solutions and making them happen. Demonstrates a disciplined approach to execution, harnessing resources effectively.
Complexity	Performs a range of work, sometimes complex and non-routine, in a variety of environments. Applies a methodical approach to issue definition and resolution. Undertakes all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools.	Drives solutions – with strong goal focused and appropriate level of urgency.
Influence	Has working level contact with customers, suppliers and partners. Interacts with and influences colleagues.	Internally – works alone, 1:1, in a team and with colleagues at all levels. Externally – works with customers, suppliers, and partners in a variety of situations.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Proficiency standard	Work activities demonstrating expected level of proficiency	Work activities demonstrating competence beyond the minimum expected
	May supervise others or make decisions which impact the work assigned to individuals or phases of projects. Makes decisions which influence the success of projects and team objectives.	Reads situation, adapts behaviours, and communicates appropriately for the situation and the audience. Actively works with others and leads by example.
Autonomy	Works under general direction. Uses discretion in identifying and responding to complex issues and assignments. Usually receives specific instructions and has work reviewed at frequent milestones. Determines when issues should be escalated to a higher level.	Can be trusted to deliver, perform and behave professionally, manages and delivers against expectations, proactively updates colleagues and behaves in line with the values and business ethics.

Software Developer Apprentice Templates

The following templates are designed to support the training provider, and will take them from training and development planning, through to the end point assessment readiness check. As with the tables above, they can be used by the training provider to help them manage the process through to completion, although training providers may also substitute their own processes and documentation as they see fit in order to effectively manage their programme.

Software Developer Template 1 – Training and Development Plan

Apprentice details

Name	
ULN	

Employer details

Company name	
Company address	
Contact name	

Training provider details

Company name	
Company address	
Contact name	

Role mapping against the software developer standard

For each area of technical and behavioural competence an overall evaluation should be provided on a three-point scale to show how often this competence is required during the normal work carried out by the employer:

- **critical** – this competence is applied most of the time;
- **desirable** – this competence is applied some of the time;
- **occasional** – this competence is rarely required.

This evaluation could form the basis of an ongoing review with the apprentice on a regular basis.

Workplace competence map

The template shows the type of activities that are identified in the apprenticeship standard for software developers as demonstrating the required competencies being applied in the workplace.

It is recognised that there are differences between the types of work carried out by different employers so this template provides the opportunity to include any other activity that demonstrates the apprentice's competence during their normal duties.

The tables below could be used to make an evaluation of the apprentice's work environment and detail the work activities that a competent apprentice should be able undertake. This activity should then lead to a discussion to identify any gaps with the employer and make a plan to redress the balance.

In the normal course of work, is the apprentice required to:	Critical	Desirable	Occasional
Write good quality code (logic) with sound syntax in at least one language	☐	☐	☐
Can develop effective user interfaces for at least one channel	☐	☐	☐
Can effectively link code to the database / data sets	☐	☐	☐
Can test code and analyse results to correct errors found using either V-model manual testing and / or using unit testing	☐	☐	☐
Can apply structured techniques to problem solving, can debug code and can understand the structure of programs in order to identify and resolve issues	☐	☐	☐
Can create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards	☐	☐	☐
Can understand and create basic analysis artefacts, such as use cases and / or user stories	☐	☐	☐

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Can understand and utilise skills to build, manage and deploy code into enterprise environments	☐	☐	☐
Can operate at all stages of the software development lifecycle, with increasing breadth and depth over time, with initial focus on build and test	☐	☐	☐
Can apply good practice approaches according to the relevant paradigm (for example object-oriented, event driven or procedural)	☐	☐	☐
Can interpret and follow: • software designs and functional/technical specifications • company defined coding standards or industry good practice for coding • testing frameworks and methodologies • company, team or client approaches to continuous integration, version and source control	☐	☐	☐
Can respond to the business environment and business issues related to software development	☐	☐	☐
Can operate effectively in their own business's, their customers' and the industry's environments	☐	☐	☐
Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures)	☐	☐	☐
Please add any other activities you think demonstrates the apprentice's competence in this area.	☐	☐	☐

What is your overall evaluation of the apprentice's opportunity to demonstrate the technical competencies in the employer's normal workplace environment?

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Knowledge module training plan

The knowledge standards define learning that should take place during the apprenticeship, **both through the training provider activities and the apprentice's independent learning**. The training provider should work with the employer to identify appropriate training for the apprentice to meet the requirements of the standard and the employer should identify opportunities within the scope of their normal business activities for the apprentice to demonstrate what they have learnt.

Knowledge and understanding will be delivered by a combination of BCS qualifications and vendor certifications in accordance with the software developer standard.

One recognised vendor or professional certification must be passed, which may be used to exempt one of the knowledge modules. Details of these, and the knowledge module that can be exempted, are contained in the standard. The training provider and the employer should agree which is best suited to their requirements.

Training plan – knowledge

BCS qualification	Selected Y/N	Vendor certification alternative chosen
BCS Level 4 Certificate in Software Development Methodologies		BCS Systems Development Essentials
BCS Level 4 Certificate in Software Languages		Cloud certified developer apache Hadoop C++ PHP Drupal Oracle SQL Developer Oracle Java Certified MCP.net MTA / MCP programming in HTML5 with JavaScript and CSS3 C#

Technical competence development plan

The following template may be used to describe how to ensure that the apprentice will be given the opportunity to demonstrate each of the required technical competencies stated in the software developer standard.

Competency requirement to meet the standard	How will this be ensured?	Responsibility (employer or training provider)?
Write good quality code (logic) with sound syntax in at least one language		
Can develop effective user interfaces for at least one channel		
Can effectively link code to the database / data sets		
Can test code and analyse results to correct errors found using either V-model manual testing and / or using unit testing		
Can apply structured techniques to problem solving, can debug code and can understand the structure of programs in order to identify and resolve issues		
Can create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards		
Can understand and create basic analysis artefacts, such as use cases and / or user stories		
Can understand and utilise skills to build, manage and deploy code into enterprise environments		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Can operate at all stages of the software development lifecycle, with increasing breadth and depth over time, with initial focus on build and test		
Can apply good practice approaches according to the relevant paradigm (for example object-oriented, event driven or procedural)		
Can interpret and follow: • software designs and functional/technical specifications • company defined coding standards or industry good practice for coding • testing frameworks and methodologies • company, team or client approaches to continuous integration, version and source control		
Can respond to the business environment and business issues related to software development		
Can operate effectively in their own business's, their customers' and the industry's environments		
Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures)		
Please add any other activity you think demonstrates the apprentice's competence in this area.		

Professional development activities plan

BCS has defined a number of professional development activities that support wider professional and career development. These activities have been associated with the various levels of responsibility, and the activities listed in the table below represent those that are appropriate for a software developer apprentice.

Training providers may wish to engage in assisting the apprentice in some of these activities as they can contribute towards the portfolio of evidence. The recommended activities include those shown below.

Professional development activities	Appropriate to the role	Agreed with apprentice and employer
Participating in group activities inside or outside the working environment that can assist with the development of interpersonal skills	☐	☐
Undertaking unpaid activities that can help to develop professional skills or offer additional insight into or understanding of their working role	☐	☐
Undertaking learning in subjects relevant to but not directly related to their role (e.g. mentoring skills, cultural awareness and diversity training), perhaps through self-study or evening classes	☐	☐
Gaining basic knowledge of the employing organisation, its business, structure, culture, products/services, operations and terminology	☐	☐
Gaining knowledge of IT activities in the employing organisation external to their function	☐	☐
Exploring a topic that is not part of their normal responsibilities, and presenting findings to colleagues and/or management	☐	☐
Attending meetings, seminars and workshops organised by a professional body and reading published material such as journals and web content	☐	☐
Undertaking learning and practice in the techniques of team and collaborative working. Gaining an understanding of the underlying concepts	☐	☐
Undertaking learning and practice in oral and written communications, including report writing and presentations	☐	☐

Software Developer Template 2 – Weekly Diary

Week number	Activities completed	Competencies displayed	Supporting evidence

Software Developer Template 3 – Periodic Workplace Competence Assessment and Remedial Action Plan

This template can be used to track the competencies being applied in the workplace on a continual/periodic basis. The training provider can then discuss any gaps with the employer and make a plan to redress the balance.

Competence assessment

In the normal course of work, is the apprentice demonstrating these competencies:	Sufficiently applied to meet minimum competence standard	Start/stop/continue – what should the apprentice start, stop or continue doing in order to develop this competence?
Write good quality code (logic) with sound syntax in at least one language	☐	
Can develop effective user interfaces for at least one channel	☐	
Can effectively link code to the database / data sets	☐	
Can test code and analyse results to correct errors found using either V-model manual testing and / or using unit testing	☐	
Can apply structured techniques to problem solving, can debug code and can understand the structure of programs in order to identify and resolve issues	☐	
Can create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards	☐	
Can understand and create basic analysis artefacts, such as use cases and / or user stories	☐	
Can understand and utilise skills to build, manage and deploy code into enterprise environments	☐	

Can operate at all stages of the software development lifecycle, with increasing breadth and depth over time, with initial focus on build and test	☐	
Can apply good practice approaches according to the relevant paradigm (for example object-oriented, event driven or procedural)	☐	
Can interpret and follow: • software designs and functional/technical specifications • company defined coding standards or industry good practice for coding • testing frameworks and methodologies • company, team or client approaches to continuous integration, version and source control	☐	
Can respond to the business environment and business issues related to software development	☐	
Can operate effectively in their own business's, their customers' and the industry's environments	☐	
Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures)	☐	
Please add any other activities you think demonstrates the apprentice's competence in this area.	☐	

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Remedial action plan

An important function of the training provider is to act as an advisor to the apprentice and the employer to ensure that the programme remains on track and any concerns are addressed. The training provider should agree how best to provide ongoing assistance/advice throughout the apprenticeship, possibly as part of their contract/service agreement with the apprentice's employer.

If any remedial action is required, the table below could be used to record it.

Remedial action (if any) required to deliver the plan/service level agreement (SLA) agreed with the employer and apprentice to demonstrate the technical competencies in the employer's normal workplace environment
Please continue on a separate sheet as required

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Software Developer – The Employer Reference Overview

This template and guidance will assist the training provider in supporting the employer when completing the employer reference, which forms a key part of the final end point assessment.

This employer reference template should be used to record the employer's comments against the grading minimum standards, criteria and dimensions, as set out in the software developer standard.

For each area of technical competence and behavioural proficiency, the employer will be asked to provide an overall evaluation on a three-point scale:

- **not met** – they have not observed this behaviour in the apprentice;
- **met** – they have observed this behaviour in the apprentice most of the time;
- **exceeded** – they have observed this behaviour in the apprentice all of the time.

They should perform an evaluation using the checkboxes, and then provide an overall evaluation of the apprentice's competence or proficiency.

The template shows the type of activities that could demonstrate the required competencies and behaviours being applied in the workplace. There are always differences between individual employers and their requirements so there is the opportunity for the employer to include any other activity that they think demonstrates the apprentice's competence. It should be completed by a senior member of the team, who is able to comment directly on work activities.

Software Developer Template 4 – The Employer Reference

Apprentice details

Name	
ULN	

Training provider details

Company name	
Company address	
Contact name	

Employer details

Name	
Company address	

Signed by:

Print name:

Job title:

Date:

Section 1

Technical competence evaluation

Please provide your evaluation of the technical competence of the apprentice using the tables below. Under each heading is a list of activities that a competent apprentice should be able to demonstrate.

Please indicate your assessment of each competence using the checkboxes, and then provide an overall evaluation of the apprentice's technical competence

Competence – Logic

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Write good quality code (logic) with sound syntax in at least one language?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in logic?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.
Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – User Interface

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can develop effective user interfaces for at least one channel?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in user interface?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Data

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can effectively link code to the database / data sets?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in data?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Test

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can test code and analyse results to correct errors found using either V-model manual testing and / or using unit testing?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in test?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.
Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Problem Solving

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can apply structured techniques to problem solving, can debug code and can understand the structure of programmes in order to identify and resolve issues?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in problem solving?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Design

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in design?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Analysis

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can understand and create basic analysis artefacts, such as user cases and / or user stories?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in analysis?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Deployment

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can understand and utilise skills to build, manage and deploy code into enterprise environments?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in deployment?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Development Lifecycle

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can operate at all stages of the software development lifecycle, with increasing breadth and depth over time with initial focus on build and test?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in development lifecycle?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Good Practice

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can apply good practice approaches according to the relevant paradigm (for example object oriented, event driven or procedural)?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in good practice?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Competence – Interpret and Follow

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can interpret and follow: • software designs and functional / technical specifications; • company defined 'coding standards' or industry good practice for coding; • testing frameworks and methodologies; • company, team or client approaches to continuous integration, version and source control?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in interpret and follow?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.   Please continue on a separate sheet if required.

Competence – Responding to Business Issues

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can respond to the business environment and business issues related to software development?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in responding to business issues?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.
Please continue on a separate sheet if required.

Competence – Operating in Different Environments

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can operate effectively in their own business's, their customers' and the industry's environments?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in operating in different environments?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.
Please continue on a separate sheet if required.

Competence – Maths

In your view, is the apprentice competent to:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures)?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's competence in maths?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of competence in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Section 2

Behaviours, business skills and level of responsibility evaluation

Please provide an evaluation as to the level of responsibility of the apprentice you are providing a reference for using the tables below. Under each heading is a list of proficiencies that a competent apprentice should display. Please indicate your assessment of the apprentice's proficiency using the checkboxes, and then provide an overall evaluation of the apprentice's proficiency.

Proficiency – Business Skills

In your view, is the apprentice proficient at:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Demonstrating an analytical and systematic approach to issue resolution?	☐	☐	☐
Taking the initiative in identifying and negotiating appropriate personal development opportunities?	☐	☐	☐
Demonstrating effective communication skills?	☐	☐	☐
Contributing fully to the work of teams?	☐	☐	☐
Planning, scheduling and monitoring own work (and that of others where applicable) competently within limited deadlines and according to relevant legislation, standards and procedures?	☐	☐	☐
Appreciating the wider business context, and how their role relates to other roles and to the business of the employer or client?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice.

What is your overall evaluation of the apprentice's business skills?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of proficiency in this area.

Please continue on a separate sheet if required.

Proficiency – Complexity

In your view, is the apprentice proficient at:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Performing a range of work, sometimes complex and non-routine, in a variety of environments?	☐	☐	☐
Applying methodical approaches to issue definition and resolution?	☐	☐	☐
Undertaking all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice

What is your overall evaluation of the apprentice's proficiency at handing complexity?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of proficiency in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Proficiency – Autonomy

In your view, is the apprentice proficient at:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Working under general direction?	☐	☐	☐
Using discretion in identifying and responding to complex issues and assignments?	☐	☐	☐
Usually receiving specific instructions and has work reviewed at frequent milestones?	☐	☐	☐
Determining when issues should be escalated to a higher level?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice

What is your overall evaluation of the apprentice's proficiency to work autonomously?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of proficiency in this area.

Please continue on a separate sheet if required.

Proficiency – Influence

In your view, is the apprentice proficient at:	The apprentice has MET this requirement	The apprentice has EXCEEDED this requirement	The apprentice has NOT MET this requirement
Interacting with and influencing colleagues?	☐	☐	☐
Having working level contact with customers, suppliers and partners?	☐	☐	☐
Supervising others or make decisions which impact the work assigned to individuals or phases of projects?	☐	☐	☐
Making decisions which influence the success of projects and team objectives?	☐	☐	☐

- **Met** – you have observed this behaviour in the apprentice most of the time.
- **Exceeded** – you have observed this behaviour in the apprentice all of the time.
- **Not Met** – you have not observed this behaviour in the apprentice

What is your overall evaluation of the apprentice's ability to influence?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of proficiency in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Section 3

Professional development

A number of professional development activities have been identified as part of the SFIA*plus* framework to help career development. These activities have been associated with the various levels of responsibility, and the activities listed in the table below represent those that are appropriate for a software developer apprentice.

In your view, is the apprentice undertaking any of the following professional development activities:	The apprentice is demonstrably undertaking this activity	The apprentice is NOT demonstrably undertaking this activity
Participating in group activities inside or outside the working environment that can assist with the development of interpersonal skills?	☐	☐
Undertaking pro bono (unpaid) activities that can help to develop professional skills or offer additional insight into or understanding of their working role?	☐	☐
Undertaking learning in subjects relevant to but not directly related to their role (e.g. foreign language courses, mentoring skills, cultural awareness and diversity training), perhaps through self-study or evening classes?	☐	☐
Gaining basic knowledge of the employing organisation, its business, structure, culture, products/services, operations and terminology?	☐	☐
Gaining knowledge of IT activities in the employing organisation external to their function?	☐	☐
Exploring a topic that is not part of their normal responsibilities, and presenting findings to colleagues and/or management?	☐	☐
Attending meetings, seminars and workshops organised by a professional body and reading published material such as journals and web content?	☐	☐
Undertaking learning and practice in the techniques of team and collaborative working. Gaining an understanding of the underlying concepts?	☐	☐
Undertaking learning and practice in oral and written communications, including report writing and presentations?	☐	☐

What is your overall evaluation of the apprentice's ability to undertake wider professional development?

Please give reasons, together with supporting examples, why you think the apprentice has demonstrated this level of proficiency in this area.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Section 4

Overall impressions and constructive feedback

This section is an opportunity for you to provide written feedback outside the rigid competency structure.

It is a free text field to allow you to share general thoughts on the apprentice's performance in case you were unable to say everything you wanted to say using the structured template. For example, you may want to highlight some of the areas where you have not been able to give the apprentice the exposure they would have liked.

We would welcome any general constructive development advice you may wish to give.

Please continue on a separate sheet if required.

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Software Developer Template 5 – Declaration and Evidence Checklists for the Completion of the Summative Portfolio

These templates are provided to support the training provider in working with the apprentice and employer to ensure the successful completion of the summative portfolio.

The checklists can be used by training providers to help them manage the process through to completion, although training providers may also substitute their own processes and documentation as they see fit.

The key responsibilities of the apprentice in producing their summative portfolio can be found in the General Guidance for Apprentices, Employers and Training Providers, as can generic guidance on how to select evidence to compile the summative portfolio.

The apprentice should gather artefacts and record information that can evidence their activities undertaken in the workplace. The portfolio of evidence should demonstrate the full range of competencies, as shown in this template, which are required by the standard to show that the apprentice can fulfil the role of a software developer.

Summative Portfolio Declaration

Apprentice declaration

Name	[first name] [surname]
ULN	[e.g.123456]
Declaration	[I confirm that all the evidence submitted is my own work and it has been completed as specified]
Signature	
Date	

Line manager declaration (employer)

Name	[line manager name]
Company	[employer name]
Declaration	I confirm that the work contained within this portfolio has, to the best of my knowledge, been completed solely by [apprentice's name]
Signature	
Date	

Training provider declaration (training provider)

Name	[observer name]
Company	[training provider name]
Declaration	I confirm that the work contained within this portfolio has, to the best of my knowledge, been completed solely by [apprentice's name]
Signature	
Date	

Information contained within this document has been republished under the terms of the Open Government Licence v3.0 © Crown copyright (2018)
Copyright © BCS 2018
Skills Framework for the Information Age © SFIA Foundation 2003, 2005, 2008, 2011, 2015
SFIA*plus* © The British Computer Society 2004, 2006, 2008, 2011, 2015
Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Page **69** of **92**

Software developer technical competencies evidence checklist

The defined competence areas and associated typical evidence are listed in this table. Not all employer businesses are identical so there will always be variation in the types of activity that will be carried out in the course of each apprentice's daily work; however, each software development apprentice must be able to demonstrate evidence of every competence.

Competence		
Logic Write good quality code (logic) with sound syntax in at least one language.		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Apprentices can write code to achieve the desired functionality and which is easy to read and understand, with good naming, indentation and commenting, and applying the fundamentals of good coding: • development paradigms (where this is object oriented programming this must include inheritance, abstractions, encapsulation, polymorphism) • software programming languages • software development tools (IDEs) • writing programs and methods • language-specific idioms • logic and flow-of-control		
Apprentices can apply: • Elements of programming – variables, assignment statements, data types, conditionals, loops, arrays, and input/output. • Functions - modular programming dividing a program into components that can be independently debugged,		

maintained, and reused writing at least two reusable functions. • Algorithms and data structures - classical algorithms for sorting and searching, and fundamental data structures**.**		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competence		
User interface Can develop effective user interfaces for at least one channel		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Evidence showing that you can apply the fundamental concepts of human–computer interaction or user experience design to program a high-quality graphical user interface.		
Can interact with screen or UI designers to ensure the logic layer integrates with the user interface		
Can develop user interface coding and implementation - techniques for building user interfaces – for at least one channel		
Can interact with testers to optimise the user interface		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competence		
Data Can effectively link code to the database / data sets		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Apprentices can link to a range of database types and embed data queries within their code		
Can make a connection to a database		
Can execute CRUD statements on the database		
Can use one-off queries and stored procedures		
Can transform returned data in to format the application requires		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competence		
Test Can test code and analyse results to correct errors found using either V-model manual testing and / or using unit testing		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Apprentices can test and analyse their code to identify errors as soon as possible in the coding process and on an interactive basis		
Can apply test and debugging strategies		
Can design and develop manual or unit tests		
Can test code segment functionality against requirements		
Can assess test results against expected results and acceptance criteria		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competence		
Problem Solving Can apply structured techniques to problem solving, can debug code and can understand the structure of programmes in order to identify and resolve issues		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Can use a minimum of two problem solving tools and techniques to identify and resolve programming issues		
Can apply structured problem solving methods		
Can apply problem-solving techniques to programming activities		

Competence		
Design Create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Can take a high-level design and can interpret and convert the design in to simple data models and/or programme modules to communicate it to others		
Can apply software design methodologies (e.g., structured or object-oriented)		
Can use standard design notation such as UML		
Can apply data modelling		
Can apply reconcile design against analysis models		
Can design software solutions to meet requirements		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competence		
Analysis Can understand and create basic analysis artefacts, such as use cases and / or user stories		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Can take a variety of data and business requirements and convert them in to basic analysis artefacts to understand and can clarify the intended use of the proposed software		
Can identify and represent required functionality (e.g. use cases)		
Can identify and represent activity workflow (e.g. activity diagrams)		

Competence		
Deployment Can understand and utilise skills to build, manage and deploy code into enterprise environments		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Can package and build completed programmes as appropriate to the resources available for deployment and for migration to different environments, including: • developing appropriate user documentation • planning for user training • data migration.		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competence		
Development life cycle Can operate at all stages of the software development life cycle, with increasing breadth and depth over time, with initial focus on build and test		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Apprentices can operate as software developers showing a good understanding of the other phases of the software development lifecycle and the deliverables that are produced at each stage and as relevant to the development methodology (waterfall, agile, test led etc)		
The apprentice can also operate in the support and maintenance phases		
Can advise third line support for relevant applications		
Can fix bugs and deal with change requests		

Competence		
Interpret and Follow Can interpret and follow: • software designs, functional/technical specifications, • company defined 'coding standards' or industry good practice for coding, • testing frameworks and methodologies, and company, team or client approaches to continuous integration, version and source control		

Minimum expected requirement	List the evidence in the portfolio that fulfils this requirement	Reflections on applying knowledge learnt
Apprentices can adapt to the employer's domain and context for software development and interpret and follow the software development approach being implemented		
Can read software designs and functional/technical specifications, especially those based upon the employer domain and context		
Can identify, interpret and follow 'coding standards'		
Can identify, interpret and follow best practice coding approaches for specific paradigms and languages		
Can identify, interpret and follow company, team or client approaches to continuous integration, version and source control		

Competence		
Respond to Business Environment Can respond to the business environment and business issues related to software development		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Apprentices can apply the following considerations when working on projects: • Business context • Business drivers (efficiency gains, increased functionality and improved quality of outputs)		

Digital Industries Apprenticeship: Standard Specific Guidance for Training Providers – Software Developer

Competence		
Industry Environment Apprentices can operate effectively in their own businesses, their customers, and the industries environments		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Undertakes all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools.		

Competence		
Application of mathematics Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures)		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Applies the maths required during Software Development Projects.		

Generic levels of responsibility evidence checklist

Areas of responsibility and associated typical evidence are shown below.

Proficiency		
Business skills Demonstrates an analytical and systematic approach to issue resolution. Takes the initiative in identifying and negotiating appropriate personal development opportunities. Demonstrates effective communication skills. Contributes fully to the work of teams. Plans, schedules and monitors own work (and that of others where applicable) competently within limited deadlines and according to relevant legislation, standards and procedures. Appreciates the wider business context, and how their role relates to other roles and to the business of the employer or client.		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Evidence that you can demonstrate an analytical and systematic approach to issue resolution.		
Evidence that you can take the initiative in identifying and negotiating appropriate personal development opportunities.		
Evidence that you can demonstrate effective communication skills.		
Evidence that you can contribute fully to the work of teams.		
Evidence that you can plan, schedule and monitor own work (and that of others where applicable) competently within limited deadlines and according to relevant legislation, standards and procedures.		
Evidence that you can appreciate the wider business context, and how your role relates		

to other roles and to the business of the employer or client.		

Proficiency		
Complexity Performs a range of work, sometimes complex and non-routine, in a variety of environments. Applies a methodical approach to issue definition and resolution. Undertakes all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools.		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Evidence that you can perform a range of work, sometimes complex and non-routine, in a variety of environments.		
Evidence that you can apply a methodical approach to issue definition and resolution.		
Undertakes all work in accordance with agreed safety, technical and quality standards, using appropriate methods and tools.		

Proficiency		
Autonomy Makes decisions which influence the success of projects and team objectives. Works under general direction. Uses discretion in identifying and responding to complex issues and assignments. Usually receives specific instructions and has work reviewed at frequent milestones. Determines when issues should be escalated to a higher level.		
Minimum expected requirement	**List the evidence in the portfolio that fulfils this requirement**	**Reflections on applying knowledge learnt**
Evidence that you can make decisions which influence the success of projects and team objectives.		
Evidence that you can operate under general direction.		
Evidence that you can use discretion in identifying and responding to complex issues and assignments.		
Evidence that you usually receive specific instructions and has work reviewed at frequent milestones.		
Evidence that you can determine when issues should be escalated to a higher level.		

Proficiency		
Influence Interacts with and influences colleagues. Has working level contact with customers, suppliers and partners. May supervise others or make decisions which impact the work assigned to individuals or phases of projects. Makes decisions which influence the success of projects and team objectives.		

Minimum expected requirement	List the evidence in the portfolio that fulfils this requirement	Reflections on applying knowledge learnt
Evidence that you can interact with and influence colleagues.		
Evidence that you have working level contact with customers, suppliers and partners.		
Evidence that you may supervise others or make decisions which impact the work assigned to individuals or phases of projects.		
Evidence that you can make decisions which influence the success of projects and team objectives.		

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Professional development activities evidence checklist

Areas of additional professional development activities that might be undertaken and associated typical evidence are shown below.

Professional development topic	Objectives	Typical evidence
Understanding organisation	Gaining basic knowledge of the employing organisation, its business, structure, culture, products/services, operations and terminology. Gaining knowledge of IT activities in the employing organisation external to their function.	• organisation charts; • company annual reports; • company website; • documents or reports from other areas of the business.
Additional business skills	Undertaking learning and practice in the techniques of team and collaborative working. Gaining an understanding of the underlying concepts. Undertaking learning and practice in oral and written communications, including report writing and presentations. Learning from experience and mistakes and applying the lessons as part of continuous improvement.	• presentations, reports or minutes of meetings that demonstrate communication skills, report writing abilities and collaborative activities; • evidence of reviewing their work and suggesting improvements or critically appraising what they did and what they learned from it, for example project retrospectives.
External activities	Participating in group activities inside or outside the working environment that can assist with the development of interpersonal skills. Undertaking pro bono (unpaid) activities that can help to develop professional skills or offer additional insight into or understanding of their working role.	• evidence of meetings attended through continuous professional development records; • evidence of activities undertaken.
Additional learning	Undertaking learning in subjects relevant to but not directly related to their role (e.g. foreign language courses,	• evidence of learning undertaken from continuous professional development records;

	mentoring skills, cultural awareness and diversity training), perhaps through self-study or evening classes. Exploring a topic that is not part of their normal responsibilities, and presenting findings to colleagues and/or management.	• evidence of presentations given to colleagues and/or management.
Professional networking	Attending meetings, seminars and workshops organised by a professional body and reading published material such as journals and web content.	• evidence of meetings attended through continuous professional development records; • written evidence summarising learning gained from reading.

Software Developer Template 6 – End Point Assessment Readiness Check

The training provider should assess whether the apprentice has met the criteria for the end point assessment as defined in the standard.

The template below is a simple checklist that may be used.

Competence	Ready	Not ready	Comments
Write good quality code (logic) with sound syntax in at least one language	☐	☐	
Can develop effective user interfaces for at least one channel	☐	☐	
Can effectively link code to the database / data sets	☐	☐	
Can test code and analyse results to correct errors found using either V-model manual testing and / or using unit testing	☐	☐	
Can apply structured techniques to problem solving, can debug code and can understand the structure of programs in order to identify and resolve issues	☐	☐	
Can create simple data models and software designs to effectively communicate understanding of the program, following best practices and standards	☐	☐	
Can understand and create basic analysis artefacts, such as use cases and / or user stories	☐	☐	
Can understand and utilise skills to build, manage and deploy code into enterprise environments	☐	☐	
Can operate at all stages of the software development lifecycle, with increasing breadth and depth over time, with initial focus on build and test	☐	☐	

Standard Specific Guidance for Training Providers – Software Developer
V3.0 May 2018

Can apply good practice approaches according to the relevant paradigm (for example object-oriented, event driven or procedural)	☐	☐	
Can interpret and follow: • software designs and functional/technical specifications • company defined coding standards or industry good practice for coding • testing frameworks and methodologies • company, team or client approaches to continuous integration, version and source control	☐	☐	
Can respond to the business environment and business issues related to software development	☐	☐	
Can operate effectively in their own business's, their customers' and the industry's environments	☐	☐	
Can apply the maths required to be a software developer (e.g. algorithms, logic and data structures)	☐	☐	